REMARKS

This application has been reviewed in light of the Office Action dated January 4, 2007. Claims 1-10, 12-29, 30-32 and 33-48 are pending in the application. By the present amendment, claims 1, 13, 22 and 36 have been amended. Previously canceled claims 30, 31 and 32 have been reinstated. Claims 11 stands cancelled without prejudice. No new matter has been added. The Examiner's reconsideration of the rejection in view of the amendment and the following remarks is respectfully requested.

By the Office Action, claims 33 and 34 stand rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the applicant regards as the invention. Claims 33 and 34 were dependent upon a cancelled claim and lacked antecedent basis. The claims on which claims 33 and 34 are dependent have been reinstated. It is respectfully submitted that the claims now have antecedent basis and are in condition for allowance for at least the reasons stated. Reconsideration is respectfully requested.

By the Office Action, claims 1-7, 12-16, 21-26 and 36-48 stand rejected under 35 U.S.C. §102 (b) as being anticipated by Pomerene et al. (U.S. Patent No. 4,679,141) (hereinafter Pomerene).

Pomerene discloses a branch predictor, a PBHT, that is divided into two parts: a fast active area that is relatively small in size but can be searched in one cycle, and a large backup area that requires many cycles to search. The areas disclosed in Pomerene are variations of branch history tables. Each entry in the Pomerene branch history tables contains a sequence of branch and target addresses (Pomerene, Figure 18), which is described as a segment. The fast and small area, the active area, uploads a segment, which is a single BHT

entry, from the backup area every time it sends a predicted cache segment (block) address to the instruction buffer (Pomerene, Column 13, lines 24-32). The instruction buffer retrieves the next instruction in its list from this segment (block) address in the cache (Pomerene, Column 13, lines 24-32). The segment corresponding to the segment (block) address sent to the instruction buffer is deleted in the active area to make space for the newly uploaded segment (Pomerene, Column 13, lines 24-32).

The PBHT in Pomerene determines which segment to upload from the backup area by utilizing an analyzer and a stager. The analyzer examines the immediate, previously uploaded segment and chooses the segment corresponding to the exit (target) address of the previously uploaded segment. (Pomerene, Column 17, lines 32-68; Column 18, lines 1-38). "The taken branch addresses are, of course, available from the segment entry (see FIG. 18) and the entry address is the exit address from the previous segment. The job of the analyzer is to compute the exit address for each successive segment based on the entry address to that segment." (Pomerene, Column 18, lines 32-38) The stager keeps the active area filled with segments predicted by the analyzer (Pomerene, Column 18, lines 39-40). The analyzer does not reference a stored record of a sequence of BHT entries to determine the next segment uploaded into the active area.

The branch predictor disclosed in Pomerene does not disclose or suggest collecting, recording or utilizing "look ahead context" information. "Look ahead context" includes "BHT entries that were recently referenced after the cache line was referenced." (Specification, p. 34, lines 10-13; p. 36, lines 19-20 to p. 37, line 1) (emphasis added). Figure 9 in the specification illustrates an example of collected look ahead context within a meta-collector. Each BA/TA pair in a cache line represents an entry of the BHT table that was

accessed subsequent to when the cache line was referenced. The look ahead context is a record of a sequence of BHT entries. Comparison of look ahead context in Figure 9 in the specification with a segment of Figure 18 of Pomerene could lead to confusion, due to their similarity in appearance. However, a segment (Figure 18), as described in Pomerene, is completely different. The segment of Figure 18 in Pomerene represents a single entry within a BHT. The look ahead context information, as illustrated in Figure 9 in the specification, represents a record of a sequence of several BHT entries. These sequences are employed to improve predictions.

It is important to note that in the specification a BHT entry was configured to have only one branch and target address pair for simplicity's sake (Specification, p. 21). A single BHT entry may include many branch and target address pairs. The set of branch and target address pairs in a BHT entry correspond to a set of instructions obtained within a single fetch. A sequence of BHT entries corresponds to sets of instructions obtained in multiple fetches.

Thus, Pomerene does not disclose or suggest collecting, recording or utilizing "look ahead context" information. The analyzer in Pomerene predicts branch addresses based upon information contained within a single segment. The analyzer does not predict branches utilizing a record of a sequence of BHT entries.

The present principles include a meta-collector, which collects "look ahead context." Look ahead context is used to predict branch locations to be prefetched during data processing. In one aspect of the present principles, the record is stored in a meta-collector, a memory storage device, a large meta-structure, a small meta-structure, or both the meta-collector and the memory storage device (Specification, 36 lines 3-13; p. 40, lines 7-10).

Accordingly, the present principles are distinguished from Pomerene, as Pomerene does not disclose or suggest recording or referencing a sequence of BHT entries for branch prediction. Although the analyzer in Pomerene contains a sequence of segments, the sequence itself is not utilized for branch prediction. As stated above, the analyzer only utilizes information within a single segment when it provides branch prediction information to its active area.

Claim 1 was amended to clarify the present principles. Claim 1 includes, *inter alia*, a meta-collector configured to collect and record look ahead context information in the meta-information which includes at least one of spatial and temporal state information associated with access of entries in a meta-structure and a memory location, such that the meta-collector provides prefetching of history table entries to the first meta-structure based upon the look ahead context information. As discussed above, Pomerene fails to disclose or suggest at least look ahead context information and a meta-collector that records and utilizes look ahead context information, as set forth in the present claim.

Claim 13 was similarly amended to clarify the present principles. Claim 13 includes, *inter alia*, a meta-collector configured to collect and record temporally and spatially sequentially unique meta-information related to access of entries of a meta-structure, each corresponding to a cache line to enable the hierarchical meta-structure operation to provide prefetching of the meta-information entries to a fastest meta-structure level based upon look ahead context information. As stated above, Pomerene fails to disclose or suggest at least look ahead context information and a meta-collector that records and utilizes look ahead context information, as set forth in the present claim.

Claim 22 was also amended to clarify the present principles. Claim 22 includes, *inter alia*, a meta-collector which collects and records one of temporally and spatially sequentially unique meta-information related to access of entries of a meta-structure and corresponding cache lines. Pomerene does not disclose or suggest a meta-collector that collects or records a sequence of entries in a meta-structure corresponding to a cache line. The sequence of segments within the analyzer of Pomerene is transient in nature; it is changed every time a segment address is provided to the instruction buffer. Indeed, the sequence itself is not used at all in Pomerene. Moreover, the sequence of segments in Pomerene is not coupled to any cache line.

Additionally, claim 36 was amended to clarify the present principles. Claim 36 includes, *inter alia*, capturing a set of entries in a meta-structure in temporal order, the entries including information associated with each entry; and storing sub-sequences of temporal entries, which share spatial context as monolithic entities wherein each monolithic entity is associated with a particular spatial context. Pomerene does not disclose or suggest recording a sequence of entries in a meta-structure as a monolithic entity. A monolithic entity is a recorded sequence of entries of a meta-structure (Specification, p. 22, lines 5-8). Although Pomerene contains a sequence of segments within the analyzer, it does not record the sequence as a monolithic entity such as in Figure 9 in the specification.

Accordingly, for at least the reasons stated above, claims 1, 13, 22 and 36 are believed to be in condition for allowance. Claims 2-7, 12, 14-16, 21, 23-26 and 37-48 are believed to be in condition for allowance due at least to their dependencies from claims 1, 13, 22 and 36.

By the Office Action, claims 8-10, 17-20, 22-29 and 33-35 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Pomerene in view of Zuraski (U.S. Patent No. 7,024,545) (hereinafter "Zuraski").

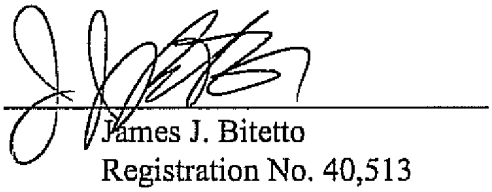
Claims 8-10 and 17-20 are believed to be in condition for allowance due at least to their dependencies from claims 1 and 13 respectfully.

With regard to claim 22, examiner cites only Pomerene as disclosing a meta-collector. As discussed above, Pomerene does not disclose or suggest a meta-collector that collects or records a sequence of entries in a meta-structure corresponding to a cache line.

Accordingly, claim 22 is believed to be in condition for allowance. Moreover, claims 23-29 and 33-35 are believed to be in condition for allowance due at least to their dependencies from claim 22. Reconsideration of the rejection is respectfully requested.

It is believed that no additional fees or charges are currently due. However, in the event that any additional fees or charges are required at this time in connection with the application, they may be charged to applicant's IBM Deposit Account No. 50-0510.

Respectfully submitted,

Date: 4/4/07 By: 

James J. Bitetto
Registration No. 40,513

Mailing Address:

KEUSEY, TUTUNJIAN & BITETTO, P.C.
20 Crossways Park North, Suite 210
Woodbury, NY 11797
Tel: (516) 496-3868
Fax: (516) 496-3869